Attorney Docket No.: 17887-004900US
Client Reference No.: SpamGuard

PATENT APPLICATION

PROCESSING OF UNSOLICITED BULK ELECTRONIC MAIL

Inventors:

Brian R. Woods, a citizen of United States, residing at,
260 Wallet Street
San Francisco, CA 94102

Udi Manber, a citizen of United States, residing at,
883 Robb Road
Palo Alto, CA 94306

Assignee:

Yahoo! Inc.
3420 Central Expressway
Santa Clara, CA 95051

Entity: Other than a small entity

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, 9th Floor
San Francisco, California 94111-3834
Tel: 403-571-4000

PATENT
Attorney Docket No.: 17887-004900US
Client Reference No.: SpamGuard

PROCESSING OF UNSOLICITED BULK ELECTRONIC MAIL

BACKGROUND OF THE INVENTION

This invention relates in general to electronic mail (e-mail) systems and, more specifically, to processing unsolicited e-mail distributed in bulk.

Unsolicited e-mail distributed in bulk, sometime referred to as Spam™, is the scourge of the Internet community. It is not uncommon for a user to receive ten to fifty unsolicited e-mail messages per day. Studies have shown that ten percent of all e-email traffic on the Internet is unsolicited bulk e-mail. A sender of unsolicited e-mail can purchase a list of millions of e-mail addresses from a list broker and easily distribute a message to the list for little or no cost. The cost of the unsolicited e-mail is paid by the providers of the Internet backbone and the users who pay access charges to download their e-mail. The senders of unsolicited e-mail offer services such as how to get rich quick, how to loose weight fast, hot stock tips, various pornographic web sites, and other shady "opportunities."

Preventing unsolicited e-mail from annoying users is a burgeoning industry. Internet service providers (ISPs) and e-mail application service providers (ASPs) experience subscriber attrition that is attributable to excessive amounts of unsolicited e-mail. For example, a user may switch to other ISP or e-mail ASP to experience a temporary reprieve from unsolicited e-mail. Unfortunately, the reprieve only last until the list brokers harvest the new e-mail address of the user.

Technology used to combat the efforts of unsolicited e-mailers is an ever-escalating arms race. The ISPs and e-mail ASPs will develop a new technology for detecting unsolicited e-mail broadcasts and the unsolicited e-mailers will develop techniques that renders the new technology ineffective. For example, once an unsolicited e-mail message is identified, the ISPs and e-mail ASPs search for other messages with the exact subject and block those messages. To combat this, the unsolicited e-mailers often attach a changing tag to each subject such that no two subject lines are the same in a large unsolicited e-mail broadcast. As those skilled in the art appreciate, more sophisticated techniques for detecting and blocking of unsolicited e-mail are desired.

SUMMARY OF THE INVENTION

The present invention involves detecting unsolicited electronic mail distributed in bulk. In one embodiment, a method for automatically processing electronic mail loads an electronic mail message. Non non-textual information is removed from the electronic mail message. A first portion from the electronic mail message is located and a first code smaller than the first portion and indicative of the first portion is generated. A second portion from the electronic mail message is located and a second code smaller than the second portion and indicative of the second portion is generated. The first code and the second code are stored.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of one embodiment of an e-mail distribution system;

Fig. 2 is a block diagram of an embodiment of an e-mail distribution system;

Fig. 3 is an embodiment of an unsolicited e-mail message exhibiting techniques used by unsolicited mailers;

Fig. 4A is a first portion of a flow diagram of an embodiment of an e-mail processing method;

Fig. 4B is an embodiment of a second portion of the embodiment of Fig. 4A;

Fig. 4C is another embodiment of a second portion of the embodiment of Fig. 4A; and

Fig. 5 is a flow diagram of an embodiment for detecting similar e-mail messages.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

The present invention processes electronic mail (e-mail) to detect unsolicited e-mail distributed in bulk. Similar messages are detected in a robust manner such that the attempts by unsolicited e-mailers to vary the text of messages in a broadcast are rendered ineffective.

Referring first to Fig. 1, a block diagram of one embodiment of an e-mail distribution system 100 is shown. Included in the distribution system 100 are an unsolicited mailer 104, the Internet 108, a mail system, and a user 116. The Internet 108 is used to connect the unsolicited mailer 104, the mail system 112 and the user, although, direct connections or other wired or wireless networks could be used in other embodiments.

The unsolicited mailer 104 is a party that sends e-mail indiscriminately to thousands and possibly millions of unsuspecting users 116 in a short period time. Usually, there is no preexisting relationship between the user 116 and the unsolicited mailer 104. The unsolicited mailer 104 sends an e-mail message with the help of a list broker. The list broker provides the e-mail addresses of the users 116, grooms the list to keep e-mail addresses current by monitoring which addresses bounce and adds new addresses through various harvesting techniques.

The unsolicited mailer provides the e-mail message to the list broker for processing and distribution. Software tools of the list broker insert random strings in the subject, forge e-mail addresses of the sender, forge routing information, select open relays to send the e-mail message through, and use other techniques to avoid detection by conventional detection algorithms. The body of the unsolicited e-mail often contains patterns similar to all e-mail messages broadcast for the unsolicited mailer 104. For example, there is contact information such as a phone number, an e-mail address, a web address, or postal address in the message so the user 116 can contact the unsolicited mailer 104 in case the solicitation triggers interest from the user 116.

The mail system 112 receives, filters and sorts e-mail from legitimate and illegitimate sources. Separate folders within the mail system 112 store incoming e-mail messages for the user 116. The messages that the mail system 112 suspects are unsolicited mail are stored in a folder called "Bulk Mail" and all other messages are stored in a folder called "Inbox." In this embodiment, the mail system is operated by an e-mail application service provider (ASP). The e-mail application along with the e-mail messages are stored in the mail system 112. The user 116 accesses the application remotely via a web browser without installing any e-mail software on the computer of the user 116. In alternative embodiments, the e-mail application could reside on the computer of the user and only the e-mail messages would be stored on the mail system.

The user 116 machine is a subscriber to an e-mail service provided by the mail system 112. An internet service provider (ISP) connects the user machine 116 to the Internet. The user activates a web browser and enters a universal resource locator (URL) which corresponds to an internet protocol (IP) address of the mail system 112. A domain name server (DNS) translates the URL to the IP address, as is well known to those of ordinary skill in the art.

With reference to Fig. 2, a block diagram of an embodiment of an e-mail distribution system 200 is shown. This embodiment includes the unsolicited mailer 104, the Internet 108, the mail system 112, and a remote open relay list 240. Although not shown, there are other solicited mailers that could be businesses or other users. The user 116 generally welcomes e-mail from solicited mailers.

E-mail messages are routed by the Internet through an unpredictable route that "hops" from relay to relay. The route taken by an e-mail message is documented in the e-mail message header. For each relay, the IP address of that relay is provided along with the IP address of the previous relay. In this way, the alleged route is known by inspection of the message header.

The remote open relay list 240 is located across the Internet 108 and remote to the mail system 112. This list 240 includes all know relays on the Internet 108 that are misconfigured or otherwise working improperly. Unlike a normal relay, an open relay does not correctly report where the message came from. This allows list brokers and unsolicited mailers 104 to obscure the path back to the server that originated the message. This subterfuge avoids some filters of unsolicited e-mail that detect origination servers that correspond to known unsolicited mailers 104 or their list brokers.

As first described above in relation to Fig. 1, the mail system 112 sorts e-mail messages and detects unsolicited e-mail messages. The mail system 112 also hosts the mail application that allows the user to view her e-mail. Included in the mail system 112 are one or more mail transfer agents, an exemplar database 208, user mail storage 212, an approved list 216, a block list 244, a key word database 230, a local open relay list 220, a short-term small message exemplars (STSME) store 224, a short-term large message exemplars (STLME) store 228, a long-term small message exemplars (LTSME) store 232, and a long-term large message exemplars (LTLME) store 236.

The mail transfer agents 204 receive the email and detect unsolicited e-mail. To handle large amounts of messages, the incoming e-mail is divided among one or more mail transfer agents 204. Once the mail transfer agent 204 gets notified of the incoming e-mail message, the mail transfer agent 204 will either discard the message, store the message in the account of the user, or store the message in a bulk mail folder of the user. The remote open relay list 240, an exemplar database 208, an approved list 216, a block list 244, a key word database 230, and a local open relay list 220 are used in determining if a received e-mail message was sent from an unsolicited mailer 104.

The user mail storage 212 is a repository for e-mail messages sent to the account for the user. For example, all e-mail messages addressed to sam1f34z@yahoo.com would be stored in the user mail storage 212 corresponding to that e-mail address. The e-mail messages are organized into two or more folders. Unsolicited e-mail is filtered and sent to the bulk mail folder and other e-mail is sent by default to the inbox folder. The user 116 can configure a sorting algorithm to sort incoming e-mail into folders other than the inbox.

The approved list 216 contains names of known entities that regularly send large amounts of solicited e-mail to users. These companies are known to send e-mail only when the contact is previously assented to. Examples of who may be on this list are Amazon.com, Excite.com, Yahoo.com, Microsoft.com, etc. Messages sent by members of the approved list 216 are stored in the user mail storage 212 without checking to see if the messages are unsolicited. Among other ways new members are added to the approved list 216 when users complain that solicited e-mail is being filtered and stored in their bulk mail folder by mistake. A customer service representative reviews the complaints and adds the IP address of the domains to the approved list 216. Other embodiments could use an automated mechanism for adding domains to the approved list 216 such as when a threshold amount of users complain about improper filtering, the domain is automatically added to the list 216 without needing a customer service representative.

The block list 244 includes IP addresses of list brokers and unsolicited mailers 104 that are known to send mostly unsolicited e-mail. The current threshold for getting on the block list 244 is sending ten thousand messages in a week. A member of the approved list 216 is not also on the block list 244. When the mail transfer agent 204 connects to the relay presenting the e-mail message, a protocol-level handshaking occurs. From this handshaking process, the actual IP address of that relay is known. E-mail message connections from a member of the block list 244 are closed down without receiving the e-mail message. Once the IP address of the sender of the message is found on the block list 244, all processing stops and the connection to the IP address of the list broker or unsolicited mailer 104 is broken. The IP address checked against the block list 244 is the actual IP address resulting from the protocol-level handshaking process and is not the derived from the header of the e-mail message.

The key word database 230 stores certain terms that uniquely identify an e-mail message that contains any of those terms as an unsolicited message. Examples of these key words are telephone numbers, URLs or e-mail addresses that are used by unsolicited mailers 104 or list brokers. While processing e-mail messages, the mail transfer agent 204 screens for these key words. If a key word is found, the e-mail message is discarded without further processing.

The exemplar database 208 catalogues the various finger print stores 224, 228, 232, 236 used in the detection algorithm along with a local open relay list 220. The exemplar database 208 acts as a server providing the information in various stores 224, 228, 232, 236 to the mail transfer agent 204 during processing of an e-mail message.

The local open relay list 220 is similar to the remote open relay list 240, but is maintained by the mail system 112. Commonly used open relays are stored in this list 220 to reduce the need for query to the Internet for open relay information, which can have significant latency. Additionally, the local open relay list 220 is maintained by the mail system 112 and is free from third party information that may corrupt the remote open relay list 240.

There are four exemplar stores 224, 228, 232, 236 coupled to the exemplar database 208. Exemplars for the e-mail messages allow counting the number of times a similar message is received by the mail system. Each e-mail message is suspected by the system 112 to be unsolicited. Exemplars for the message are stored in one of the stores 224, 228, 232, 236 to serve as a fingerprint for that message. The short-term message exemplars stores 224, 228 store the most recent two hours of messages. If messages that are similar to each other are received by the short-term message exemplars stores 224, 228 in sufficient quantity, the message is moved to the long-term message exemplars stores 232, 236. The long-term message stores 232, 236 retain a message entry until no similar messages are received in a thirty-six hour period. There are two stores for each of the short-term stores 224, 228 and the long-term stores 232, 236 because there are different algorithms that produce different exemplars for long messages and short messages.

Referring next to Fig. 3, an embodiment of an unsolicited e-mail message 300 is shown that exhibits some techniques used by unsolicited mailers 104. The message 300 is subdivided into a header 304 and a body 308. The message header includes routing information 312, a subject 316, the sending party and other information. The routing information 312 is often inaccurate along with the reference sending party to avoid blocking further unsolicited messages from that source. Included in the body 308 of the message is the information the unsolicited mailer 104 wishes the user 116 to read. Typically, there is a URL 320 or other mechanism for contacting the unsolicited mailer in the body of the message in case the message presents something the user is interested in. To thwart an exact comparison of message bodies 308 to detect unsolicited e-mail, a evolving code 324 is often included in the body 308.

With reference to Figs. 4A and 4B, a flow diagram of an embodiment of an e-mail processing method is depicted. Fig. 4C is not part of this embodiment. The process starts in step 404 where the mail transfer agent 204 begins to receive the e-mail message 300 from the Internet 108. This begins with a protocol level handshake where the relay sending the message 300 provides its IP address. In step 408, a test is performed to determine if the source of the e-mail message 300 is on the block list 244. If the source of the message is on the block list 244 as determined in step 412, the communication is dropped in step 416 and the e-mail message 300 is never received. Alternatively, processing continues to step 420 if the message source is not on the block list 244.

E-mail messages 300 from certain sources are accepted without further investigation. Each message is checked to determine if it was sent from an IP addresses on the approved list 216 in steps 420 and 424. The IP addresses on the approved list 216 correspond to legitimate senders of e-mail messages in bulk. Legitimate senders of e-mail messages are generally those that have previous relationships with a user 116 where the user assents to receiving the e-mail broadcast. If the IP address is on the approved list 216, the message is stored in the mail account of the user 116.

Further processing occurs to determine if the message 300 was unsolicited if the source of the message 300 is not on the approved list 216. In step 432, the message body 308 is screened for key words 230. The key words 230 are strings of characters that uniquely identify a message 300 as belonging to an unsolicited mailer 104 and may include a URL 320, a phone number or an e-mail address. If any key words are present in the message body 308, the message 300 is discarded in step 416 without further processing.

To determine if the e-mail message 300 has been sent a number of times, an algorithm is used to determine if the e-mail message 300 is similar to others received in the past. The algorithm does not require exact matches and only requires some of the exemplars that form a fingerprint to match. In step 440, exemplars are extracted from the message body 308 to form a fingerprint for the message 308. A determination is made in step 444 as to whether there are two or more exemplars harvested from the message body 308.

In this embodiment, more than two exemplars are considered sufficient to allow matching, but two or less is considered insufficient. When more exemplars are needed, a small message algorithm is used to extract a new set of exemplars to form the fingerprint in step 448. The small message algorithm increases the chances of accepting a string of characters for generating an exemplar upon. Future matching operations depend upon whether the exemplars were extracted using the small message or large message algorithm to generate those exemplars. The small message stores 224, 232 are used with the small message algorithm, and the large message stores 228, 232 are used with the large message algorithm.

The thresholds for detection of unsolicited e-mail are reduced when the message is received by the mail system 112 from an open relay. Open relays are often used by unsolicited mailers 104 to mask the IP address of the true origin of the e-mail message 300. By masking the true origin, searching for messages from that IP address is not used to detect unsolicited mailers. However, the IP address of the relay that last sent the message to the mail system 112 can be accurately determined. The actual IP address of the last relay before the message 300 reaches the mail system 112 is known from the protocol level handshake with that relay. The actual IP address is first checked against the local open relay list 220 for a match. If there is no match, the actual IP address is next checked against the remote open relay list 240 across the Internet 108. If either the local or remote open relay lists 220, 240 include the actual IP address, first detection threshold is reduced from fifty to twenty-five, and the second detection threshold is reduced from one hundred to fifty in step 460 as described below.

Depending on whether the e-mail message 300 is a short or long message as determined in step 444, either the STSME store 224 or STLME store 228 is checked for a matching entry. The STSME and STLME stores 228, 224 hold the last two hours of message fingerprints along with a first count for each. The first count corresponds to the total number of times the mail transfer agents 204 have seen a similar message within a two hour period so long as the count does not exceed the first threshold.

A test for matches is performed in step 464. A match only requires a percentage of the exemplars in the fingerprint to match. In this embodiment, a match is found when all of the exemplars of a fingerprint stored in the respective STSME or STLME store 228, 224 are found in the exemplars of the message currently being processed. Other embodiments could only require a percentage of the exemplars in the respective STSME or STLME store 228, 224 be found in the message being processed.

If a match is found in step 464 between the current e-mail message 300 and the respective STSME or STLME store 228, 224, processing continues to step 468 where a first count is incremented. The first count is compared to the first threshold in step 472. Depending on the determination in step 456, the first threshold is either fifty or twenty-five. If the first threshold is not exceeded, processing continues to step 484 where the e-mail message 300 is stored in the user's inbox folder.

Alternatively, processing continues to step 476 if the first threshold is exceeded by the first count. The fingerprint of exemplars for the e-mail message 300 is moved from the short-term store 224, 228 to the respective long-term store 232, 236 in step 476. In step 480, the new fingerprint will replace the oldest fingerprint in the long-term store 232, 236 that has not been incremented in the last thirty-six hours. A fingerprint becomes stale after thirty-six hours without any change in count. If there is no stale entry, the new fingerprint is added to the store 232, 236 and an index that points to the fingerprint is added to the beginning of a list of indexes such that the freshest or least stale fingerprint indexes are at the beginning of the index list of the long-term store 232, 236. Once the fingerprint is added to appropriate the long-term store 232, 236, the e-mail message 300 is stored in the account of the user in step 484.

Returning back to step 464, processing continues to step 486 if there is not a match to the appropriate short-term message database 224, 228. In step 486, the message fingerprint is checked against the appropriate long-term message store 232, 236. Only a percentage (e.g., 50%, 80%, 90%, or 100%) of the exemplars need to exactly match an entry in the appropriate long-term message store 232, 236 to conclude that a match exists. The long-term message store 232, 236 used for this check is dictated by whether the long or short message algorithm is chosen back in step 444. If there is not a match determined in step 488, the e-mail message 300 is stored in the mailbox of the user in step 484. Otherwise, processing continues to step 490 where the second count for the fingerprint entry is incremented in the long-term store 232, 236. When the second count is incremented, the fingerprint entry is moved to the beginning of the long-term store 232, 236 such that the least stale entry is at the beginning of the store 232, 236.

In step 492, a determination is made to see if the e-mail message 300 is unsolicited. If the second threshold is exceeded, the e-mail message is deemed unsolicited. Depending on determination made in step 456 above, the second threshold is either one-hundred or fifty. If the second threshold is exceeded, the e-mail message 300 is stored in the bulk mail folder of the user's account in step 494. Otherwise, the e-mail message 300 is stored in the inbox folder. In this way, the efforts of unsolicited mailers 104 are thwarted in a robust manner because similar messages are correlated to each other without requiring exact matches. The first and second thresholds along with the times used to hold fingerprints in the exemplar database 208 could be optimized in other embodiments.

With reference to Figs. 4A and 4C, a flow diagram of another embodiment of an e-mail processing method is depicted. Fig. 4B is not a part of this embodiment. This embodiment checks long-term message exemplars store 232, 236 before short-term message exemplars store 224, 228.

Referring next to Fig. 5, a flow diagram 440 of an embodiment for detecting similarities between e-mail messages is shown. The process begins in step 504 where an e-mail message 300 is retrieved. Information such as headers or hidden information in the body 308 of the message 300 is removed to leave behind the visible body 308 of the message 300. Hidden information is anything that is not visible the user when reading the message such as white text on a white background or other HTML information. Such hidden information could potentially confuse processing of the message 300.

To facilitate processing, the visible text body is loaded into a string or an array in step 512. The index of the array is initialized to zero or the first element of the array. In step 516, the first twenty characters in the array are loaded into an exemplar algorithm. Although any algorithm that produces a compact representation of the twenty characters could be used, the following equation is used in step 520:

$$E_n = \left(\sum_{i=1}^{i=20} t_i p^{20-i} \right) \bmod M \qquad (1)$$

In equation 1 above, the potential exemplar, E, starting at array index, n, is calculated for each of the twenty characters, t_i, where p is a prime number and M is a constant. In this embodiment, M is 2^{32} and p is 56,7319.

Only some of the potential exemplars E resulting from equation 1 are chosen as good anchors such that the potential exemplar E is stored in the fingerprint. Further to step 520, the potential exemplar E is converted to a binary value and masked by an octal value that is also converted to binary. If the result from the masking step includes any bits equal to one, the potential exemplar *E* is used in the fingerprint for the message 300. The large message algorithm uses an octal value of 157_8 converted into a binary mask and the small message algorithm uses an octal value of 55_8 converted into a binary mask such that the small message algorithm is more likely to accept any potential exemplar *E*.

If the potential exemplar *E* is chosen as an anchor in step 524, it is added to the fingerprint and the array index is incremented by twenty in step 544. The index is incremented by twenty to get a fresh set of characters to test for an anchor. In step 548, the number of exemplars chosen for the fingerprint is tested to see if it exceeds forty. Once forty exemplars are selected from the message 300, processing continues to step 556 where the exemplars for the fingerprint are stored in descending order. Storing in descending order allows searching more efficiently through the fingerprint during the matching process. If there are less than forty exemplars, a test is performed in step 552 to see if the end of the array is reached which signals that the analysis of the message body 308 is complete. Presuming the array is not completely analyzed, processing continues to step 516 where twenty new characters are loaded and analyzed.

Alternatively, the index is only incremented by one in step 528 if the anchor is not chosen in step 524. Only a single new character is needed to calculate the next potential exemplar since the other nineteen characters are the same. The exit condition of over forty exemplars and the exit condition of passing the end of the array are checked in steps 532 and 536. If neither exit condition is satisfied, the next element from the array is loaded in step 540. A simplified equation 2 may be used to determine the next potential exemplar, E_{n+1}, by adding the last coefficient and removing the first one:

$$E_{n+1} = \left(pE_n + t_{21} - t_1 p^{19}\right) \bmod M \qquad (2)$$

In this way, the exemplars that form the fingerprint for the message body are calculated.

In light of the above description, a number of advantages of the present invention are readily apparent. E-mail messages that are similar to each other, but not exact, are detected in an efficient manner. Attempts by unsolicited mailers to send bulk mail are thwarted by robust matching of unsolicited messages to find patterns of distribution that exceed certain thresholds.

A number of variations and modifications of the invention can also be used. For example, the invention could be used by ISPs on the server-side or users on the